Notes on Stat 139: Linear models

Jimmy Qin

Fall 2019

I took Statistics 139: linear models in fall of 2019. It was lectured by Kevin Rader.

Lecture 3. t- and z-tests

Central limit theorem

Let $X_i \sim \mathcal{N}(\mu, \sigma^2)$ for $1 \leq i \leq n$ and $n \to \infty$. Then the distribution of the sample mean is

$$\bar{X} \sim \mathcal{N}(\mu, \frac{\sigma^2}{n}).$$

χ^2-distribution

Let $Z_i \sim \mathcal{N}(0, 1)$ be i.i.d. and define

$$Y = \sum_{i=1}^{k} Z_i^2.$$

Then Y follows a χ^2-distribution with k degrees of freedom, written

$$Y \sim \chi_k^2.$$

The PDF of this distribution is defined on support $x > 0$:

$$f(x) = \frac{1}{2^{k+1}\Gamma(k/2)} x^k e^{-x/2}.$$

We do we care about the χ^2-distribution? If the distribution of the sample variance of a random variable is normal, then if we have k measurements, the *total* error will be χ_k^2-distributed. More precisely, if our measurements have variance σ^2 then our sample variance is

$$S^2 = \frac{1}{n-1} \sum_{i=1}^{n} (X_i - \bar{X})^2$$

and follows a χ^2-distribution

$$\frac{n-1}{\sigma^2} S^2 \sim \chi_{n-1}^2.$$

t-distribution

Let $Z \sim \mathcal{N}(0,1)$, $Y \sim \chi^2_k$, and let Z and Y be independent. We define the t-distribution with k degrees of freedom, $T \sim t_k$, by

$$T := \sqrt{k}\frac{Z}{\sqrt{Y}}.$$

The PDF is

$$f(t) = \frac{\Gamma(\frac{k+1}{2})}{\sqrt{k\pi}\Gamma(k/2)}\left(1+\frac{t^2}{k}\right)^{-\frac{k+1}{2}}.$$

Geometrically, the t-distribution looks like the standard Normal, but has "fatter" tails. The higher the degrees of freedom, k, the more closely it looks like $\mathcal{N}(0,1)$. This is because the more degrees of freedom we have, the flatter the χ^2-distribution is.

Why do we care about the t_k-distribution? Many times, we will not know what the true variance σ^2 is, but we *will* have some idea of an ideal population mean, or perhaps a population mean from a hypothesis in a hypothesis test. Therefore, we will calculate z-scores with the *sample variance*, which is

$$T = \frac{\bar{X}-\mu}{S/\sqrt{n}}.$$

The random variable T follows a t distribution because the top, $\bar{X}-\mu$, is assumed Gaussian, and the bottom, S^2, is χ^2_k-distributed.

t-based testing, confidence intervals

One-sample t-test

We would like to test the hypothesis

$$\mathrm{H}_0 : \mu = \mu_0$$

against a one or two-sided alternative hypothesis. To do so, we will calculate the **one-sample t-statistic**, which is an analogue of the z-score where the population variance has been replaced with the sample variance,

$$t = \frac{\bar{X}-\mu}{S/\sqrt{n}}.$$

We compute the p-value by comparing this t-statistic with a t-distribution with $n-1$ degrees of freedom. A **one-sided p-value** supposes the error is in only one direction; it accounts for the area of only one tail. A **two-sided p-value** supposes the error is directionless; it accounts for the area of both tails; because the t-distribution is symmetric, the two-sided p-value is always *twice* the one-sided p-value.

One-sample confidence interval for μ

A **confidence interval for the mean** is a "range of plausible values" for the population mean, based on (1) the sample mean (2) the sample variance. We construct it as follows:

$$t\text{-based confidence interval } = \bar{X} \pm t^* \times \frac{S}{\sqrt{n}}.$$

Here, t^* is the quantile from a t-distribution with $n-1$ degrees of freedom. Basically what this says is that t^* is a number you get from a table, which is independent of your data. For example, see `http://uregina.ca/~gingrich/tt.pdf`. Then you multiply it by the sample variance and insert the $\frac{1}{\sqrt{n}}$ factor which accounts for the mean. This is the interval length.

Inference for a difference in means

The preferred method for deciding whether the means of two populations are different is called the **2-sample t-test**. We will motivate it now. Our null and alternative hypotheses are the following:

$$\mathrm{H}_0 : \mu_1 = \mu_2 \text{ and } \mathrm{H}_A : \mu_1 \neq \mu_2.$$

To see if the means are significantly different, we will test how big

$$\bar{X}_1 - \bar{X}_2$$

is compared to the standard deviation of this difference, which follows from linearity of variances:

$$\mathrm{Var}(\bar{X}_1 - \bar{X}_2) = \sqrt{\frac{\sigma_1^2}{n_1} + \frac{\sigma_2^2}{n_2}} \to \sqrt{\frac{s_1^2}{n_1} + \frac{s_2^2}{n_2}}.$$

Unsurprisingly, the **two-sample t-statistic** is

$$T = \frac{1}{\sqrt{\frac{s_1^2}{n_1} + \frac{s_2^2}{n_2}}} \times \left[(\bar{X}_1 - \bar{X}_2) - (\mu_1 - \mu_2)|_{H_0} \right].$$

If we are performing this calculation by hand, we approximate the degrees of freedom as the *smaller* of $n_1 - 1$ or $n_2 - 1$. `R` uses something called the **Satterthwaite approximation**.

Now we can obtain a sample statistic, and the rest of the mechanics proceeds just like the one-sample t-test. This is called the **nonpooled t test**, in contrast to the **pooled t test**, which assumes $\sigma_1^2 = \sigma_2^2$. You should use the nonpooled t test.

Paired data

When comparing two groups, sometimes the observations are naturally paired. For example, before vs. after treatment, or twin studies. If the data are paired, we should perform a **paired t test**. This is an extension of the 1-sample test, where we take the *differences* between the pairs and perform a 1-sample test on the differences.

The standard deviation used for this t-test is the sample standard deviation of the *set of differences* between the paired sets.

Difference in proportions

When comparing proportions between groups, the t distribution is no longer applicable. This is because in Bernoulli data, which there is really just *one* parameter to estimate – the fraction p of successes. That is because the population value of p determines both the mean and variance:

$$\mu = p, \sigma^2 = p(1-p).$$

Therefore, we can fit the distribution of $Y = \sum_i^n X_i$, where X_i are i.i.d. Bern(p), to the normal where μ and σ^2 are expressed in terms of the only unknown – p:

$$Y \sim \text{Binom}(n, p) \approx \mathcal{N}(p, np(1-p)).$$

Now we can figure out how to do the **difference between proportions test**. Let

$$\hat{p}_1 = \frac{1}{n_1} Y_1 \text{ and } \hat{p}_2 = \frac{1}{n_2} Y_2.$$

Then the difference follows the distribution

$$\hat{p}_1 - \hat{p}_2 \sim \mathcal{N}(p_1 - p_2, n_1 p_1(1-p_1) + n_2 p_2(1-p_2)),$$

since variances add. We can test hypotheses based on what we believe $(p_1 - p_2)|_{\text{H}_0}$ is in real life; for example, if we are trying to test whether there *exists* a difference, we would test $(p_1 - p_2)|_{\text{H}_0} = 0$.

More rigorously, as long as we have at least 10 successes and failures for both sets of data, we test the **two-sample z-statistic**

$$Z = \frac{(\hat{p}_1 - \hat{p}_2) - (p_1 - p_2)|_{H_0}}{\sqrt{\hat{p}_p(1-\hat{p}_p)(n_1^{-1} + n_2^{-1})}}.$$

Lecture 4. Testing errors and ANOVA

We would like to learn about errors in testing. There are two kinds of errors: **type I** and **type II**. A type I error is when H_0 is true but we falsely reject H_0. A type II error is when H_a is true but we fail to reject H_0.

Error rates

- Rate of a type I error, α: The rate of a type I error is called α. This is what we compare the p-value to; normally we take $\alpha = 0.05$. The p-value is calculated assuming the null hypohtesis is true, so if the null hypothesis is really true, we will reject it around 5% of the time.
- Rate of a type II error, β: The rate of a type II error is called β, and $1 - \beta$ is called the **statistical power** of the test. Computing this probability can be difficult. Statistical power is calculated before gathering data and has to do with (1) the distance of the calculated mean from the null hypothesis value, called **effect size** (2) the standard deviation of the sample (3) the sample size.

Multiple comparisons

Consider l similar groups; we want to see if the group means are all the same. This is the **multiple comparisons problem**.

One (not very good) way to do this is to conduct $\binom{l}{2}$ two-sample t-tests to determine which paris of groups have a significant difference. However, this leads to an *overly high probability of type I*

error, since there are more chances to make mistakes. The probability of rejecting at least one true H_0, given that H_0 is always true, is

$$1 - (1 - \alpha)^{\binom{n}{2}}$$

which increases quickly with n. Therefore, multiple comparisons with only the regular pairwise t-test leads to *significant rejections that are not truly there*, but simply by chance.

Generally, we would like the overall probability of a type I error to be some fixed value α, independent of how many groups there are.

Bonferroni correction

One solution to the multiple comparisons problem is to postulate a new level of α. This is called the **Bonferroni correction**. It is very simple and postulates the new value α^* for any one of the paired t-tests, where there are n groups to do pairwise comparisons with:

$$\alpha^* = \binom{n}{2}^{-1} \alpha.$$

This significantly increases the confidence level. In fact, the Bonferroni correction is a very *conservative* solution and adjusts the type I error too much (i.e. adjusts for the worst case scenario). There are other approaches: **Tukey**, **Scheffe**, **False discovery rate** approaches try to balance the effect on type I and type II error rates.

F distribution

Definition of F distribution

First, we present the definition of F distribution. Let $X \sim \chi^2_{d_1}$, $Y \sim \chi^2_{d_2}$ be independent. Then the ratio

$$F = \frac{X/d_1}{Y/d_2} \sim F_{d_1,d_2}$$

is said to follow the **F-distribution with d_1 and d_2 degrees of freedom**. The support is $f > 0$ and the inverse F^{-1} is a d_2, d_1-df F-distribution.

Motivating F

Why is F important? Suppose $X_i \sim \mathcal{N}(\mu_x, \sigma^2)$ and $Y_j \sim \mathcal{N}(\mu_Y, \sigma^2)$ for $1 \leq i \leq n_X$ and $1 \leq j \leq n_Y$. Then the ratio of the sample variances is F-distributed,

$$F = \frac{S_X^2}{S_Y^2} \sim F_{n_X - 1, n_Y - 1}.$$

This is because the sample variance is defined

$$S_X^2 = \frac{1}{n_X - 1} \sum_{i=1}^{n_X} (x_i - \bar{x})^2 \approx \frac{1}{n_X} \sum_{i=1}^{n_X} (x_i - \bar{x})^2 \text{ and } \sum_{i=1}^{n_X} z_i^2 \sim \chi^2_{n_X},$$

where $z_i \sim \mathcal{N}(0,1)$ are i.i.d.

The two groups X and Y *must have the same true variance*, σ^2. Actually, it seems to me that you could extend it to different variances, but you would have to renormalize one of the variances first.

The true power of F is that we will use it to compare within-group and between-group estimates of the individual variances, σ^2, in ANOVA.

ANOVA (Analysis of variance)

Let us review the kinds of hypotheses we have made for hypothesis tests. With one population,

$$H_0 : \mu = \mu_0.$$

With two populations,

$$H_0 : \mu_1 = \mu_2.$$

How about 3 or more populations? The technique in this case is called **analysis of variance**, or **ANOVA**. The null hypothesis in this case is

$$H_0 : \mu_1 = \mu_2 = \cdots = \mu_n,$$

where n is the number of populations we are interested in.

Separate two kinds of variability

With n populations, there are two kinds of variability. Let Y_{ij} be an individual value of measurement j in group i, $\bar{Y}_i$ be the **group mean for group** i, and $\bar{Y}$ be the **grand mean**, which is $\bar{Y} = \frac{\sum_{ij} Y_{ij}}{\sum_i N_i}$, where N_i is the number of measurements in group i. The two kinds of variability are

1. Variability within groups: variation of individual values around the group means, $Y_{ij} - \bar{Y}_i$.
2. Variability between groups: variation of groups means around the overall grand mean, $\bar{Y}_i - \bar{Y}$.

The idea of ANOVA is to separate the effects (1) and (2). *If the inter-group variability is large compared to the intra-group variability, the group means in the population are different.* Therefore, ANOVA determines whether variability in data is mainly from variation within groups or variation between groups.

One-way ANOVA theory

Suppose the n groups have possibility different means but all have the same variance,

$$Y_{ij} \sim \mathcal{N}(\mu_i, \sigma^2).$$

It is convenient to write

$$Y_{ij} = \mu_i + \epsilon_{ij}, \text{ where } \epsilon_{ij} \sim \mathcal{N}(0, \sigma^2).$$

Therefore, there are $n+1$ parameters: μ_i for $1 \le i \le n$, and σ^2 (one common variance).

Now we will begin the procedure of estimation, in two steps.

1. Intra-group variance: Because σ^2 is assumed to be equal for all n groups, all sample variances can be used to estimate σ^2. We will take the estimate of σ^2 to be the **mean square error (MSE)** or **mean square within groups (MSW)**:

$$\text{MSE} = \frac{\sum_i (n_i - 1) S_i^2}{\sum_i (n_i - 1)}.$$

   Here, the purpose of $(n_i - 1)S_i^2$ is to get rid of the $\frac{1}{n_i - 1}$ prefacor on the calculation of the sample standard deviation.

2. Inter-group variance: Now we do something truly **meta**! We say that μ_i, the group means, *should themselves be distributed normally.* The sampling distribution of sample means is

$$\bar{Y} \sim \mathcal{N}(\mu, \frac{\sigma^2}{n}).$$

   Therefore, *if*

$$\mu_1 = \mu_2 = \cdots = \mu_n$$

   is true, we can *also estimate* σ^2 with the group means, using a parameter called the **mean square of the model (MSM)** or **mean square between groups (MSB) or (MSG)**:

$$\textbf{MSM} = \frac{\sum_i n_i (\bar{Y}_i - \bar{Y})^2}{n_{\text{groups}} - 1}.$$

   Here, $n_{\text{groups}} = \sum_i 1$ is the number of groups.

3. To determine whether

$$\mu_1 = \mu_2 = \cdots = \mu_n$$

   is reasonable, we must compare MSM to MSE; if MSM is significantly bigger, then $\mu_1 = \mu_2 = \cdots = \mu_n$ may not be true. This calls for the F-distribution. The **ANOVA** F **statistic** is

$$F = \frac{\frac{1}{n-1} \sum_{i=1}^n n_i (\bar{Y}_i - \bar{Y})^2}{\frac{1}{(\sum_{i=1}^n n_i) - n} \sum_{i=1}^n (n_i - 1) S_i^2} = \frac{\text{MSB}}{\text{MSW}}.$$

   Let us call $N = \sum_i^n n_i$. We reject H_0 for large values of F. Specifically, if

$$F > F^*_{\alpha, n-1, N-n}$$

   or the corresponding p-value is less than α.

Lecture 5: Transformations

Often, we will use transformations to turn data on which tests are *not* applicable into data for which the tests *can* be applied. How?

Assumptions in t-based methods

Consider the 1-sample t-test. In order to perform inferences, we make two assumptions:

- Observations are independent
- Observations are normally distributed.

What happens to the procedure when these assumptions break down? For the two-sample t-test, we have *more* assumptions:

- Two groups are independent.

For the pooled t-test and ANOVA F-test, we make another major assumption:

- The two groups have the same population variance, $\sigma_1^2 = \sigma_2^2 = \sigma^2$.

Idea of robustness

The performance of an inferential procedure when the assumptions *fail* is called its statistical **robustness**. We quantify robustness, of course, with the errors – both type I and type II (i.e. $1 -$ power). We can evaluate these quantities with probability theory or numerical methods, but numerical methods are usually much easier.

Nonlinear transformation

We will use transformations to make the data in a usable form for the t or ANOVA F-tests, specifically, we would like to make the data more *nearly normal*. How?

Here are some transformations:

- Log (natural) transformation: If the initial data is labeled Y_i, define

  $$Z_i = \log Y_i.$$

  This is used when the data are very skewed to the right, and all the data are positive. Our hope is that we will get a symmetric histogram. In the case of the two-sample t-test, we hope to get *two* symmetric histograms with similar spreads but possibly different centers.

- Square root transformation: This is

  $$Z_i = \sqrt{Y_i}$$

  and is used if the measurement data are *moderately* skewed to the right.

- Reciprocal transformation: Of course, this is

  $$Z_i = Y_i^{-1}.$$

  This is used for several skewed data, such as waiting or failure times. It can be used with negative data.

- Logit transformation:
$$Z_i = \frac{Y_i}{1 - Y_i}$$
This is good for proportions, $0 \leq Y_i \leq 1$. It transforms to the real line, $Z_i \in \mathbb{R}^+$.

Nonparametric ranked tests

There is a difference between **parametric procedures**, which make assumptions about the underlying distribution of the observations, and **nonparametric procedures**, which do not make assumptions about the underlying distribution. Let us study some nonparametric methods.

Rank-sum test

An example of a nonparametric ranked test is the **rank-sum test**. Suppose we have two samples, Y_{1i} and Y_{2j} where $1 \leq i \leq n_1$ and $1 \leq j \leq n_2$. We don't want to assume normal distributions or anything like that, so we transform the samples.

Namely, we calculate the *rank* of the sample with regards to the combined dataset $\mathbf{Y} = \{Y_{1i}, Y_{2j}\}$. We define
$$Z_{1i} = \text{Rank}(Y_{1i}|\mathbf{Y}) \text{ and } Z_{2j} = \text{Rank}(Y_{2j}|\mathbf{Y}). \tag{1}$$
The test statistic is calculated from the ranks;
$$W = \sum_i Z_{1i}. \tag{2}$$
We only need to calculate W for the first sample, since it tells us about the sums of the Z_{2j} by exclusion. The null and alternative hypotheses are that the *true quantiles in the two groups are the same*, or not. If the shapes of the distributions are similar, we can make this more specfic: The *true medians in the two groups are the same*, or not.

If H_0 is true and there are no ties, the ranks are distributed
$$Z_{ki} \sim \text{DUnif}(1, 2, \cdots, n_1 + n_2). \tag{3}$$
We would like to find the distribution of W given H_0. By CLT, we can use the properties of Unif to say
$$W \sim \mathcal{N}(\mu_W, \sigma_W^2), \mu_W = \frac{n_X(n_X + n_Y + 1)}{2}, \sigma_W^2 = \frac{n_X n_Y (n_X + n_Y + 1)}{12}. \tag{4}$$
This is implemented as the **Wilcox test** in `R`. Obviously, from the above distribution, we can extract a test statistic and therefore a p-value.

Lecture 6: Resampling methods

Resampling to build reference distributions

A **reference distribution** is the distribution of a statistic. For example, we often use χ distribution or t distributions. Although we often use ready-made reference distributions, we can also

use resampling methods to generate an **empirical reference distribution** from the data.

There are two common methods:

- Data generating process: sample from a theoretical distribution (i.e. to perform simulations). This is *parametric*.
- Resampling from the observed sample (i.e. bootstrapping). This is *nonparametric*.

Let's study the bootstrapping technique. It is widely used to build empirical reference distributions, such as confidence intervals for the mean.

Bootstrapped confidence intervals

A **boostrapped sample** is obtained by treating the sample *as if* it was the population and *resampling* from the sample. For example, suppose there are 20 entries in the data $\mathbf{X}$ and we want to find a confidence interval for $\bar{X}$. Draw numbers from $\mathbf{X}$ over and over and over and generate a histogram of outcomes.

How can we use the outcomes to calculate a confidence interval? There are 2 common ways. Let $\bar{X}^*$ represent the bootstrapped sample means. Basically, if we want a sample of size 10, we need a sample from the empirical data of size 10, and then we do this sampling over and over. (Normally, we are interested in the case in which our sample is the same size as our dataset, although this is not always the case.)

- Use the $\bar{X}^*$ to calculate the standard error and use

$$\bar{X} \pm t^*_{n-1} S^* \tag{5}$$

  where $S^* = \sqrt{\frac{n}{n-1}} S(\bar{X}^*)$.

- Select the quantiles from the empirical sampling distribution $\bar{X}^*$ (i.e. cut the sampling distribution off at the 0.025 and 0.975 quantiles).

We can use the second method to find the bootstrapped quartiles, the bootstrapped estimate of the median, and the bootstrap confidence interval for a difference in means. (For the difference in means, just calculate $\bar{X}^*_1 - \bar{X}^*_2$ for each sample of the desired size, and then repeat the sampling a bunch of times. Then there is a distribution of $\bar{X}^*_1 - \bar{X}^*_2$ and you can, for example, select the range 0.025 to 0.975.)

Some notes about bootstrapping:

- We rely on independence when performing the resampling. This is an important assumption.
- We usually sample the same original sample size n, since that's what we're interested in (but not always).
- The bootstrapping technique is important only if we do not know (or have a good idea of) the original distribution.

Permutation test

The **permutation test**, like the rank-sum test, is another alternative to parametric testing. (Recall that some parametric tests are z, t, F, χ^2.) The permutation test is more interpretable and and improvement on the rank-sum test (we don't really know what the rank-sum test is estimating!).

Hypotheses of permutation tests:

- H_0: The distribution of outcomes is not related to group status.
- H_A: The distribution of outcomes is associated with group status.

The test statistic in any permutation test is the *difference in average observed outcomes* between the two groups. The outcome could be anything – mean, median, proportion, etc. Or, it could be a more complicated function.

The idea of groups identity is the key to permutation test (and is why it's called a permutation test). If the group label (0 or 1) doesn't matter, then we should able to permute the group labels without any effect on the difference in sample means, for example. Let's give an example: let the group labels be 0 and 1. The observed data are Y_{1i} and Y_{0j}, depending on which group they are in. Therefore, if H_0 is true, I should be able to switch the 0 and 1 labels and not affect the difference in means. The below figure shows this logic:

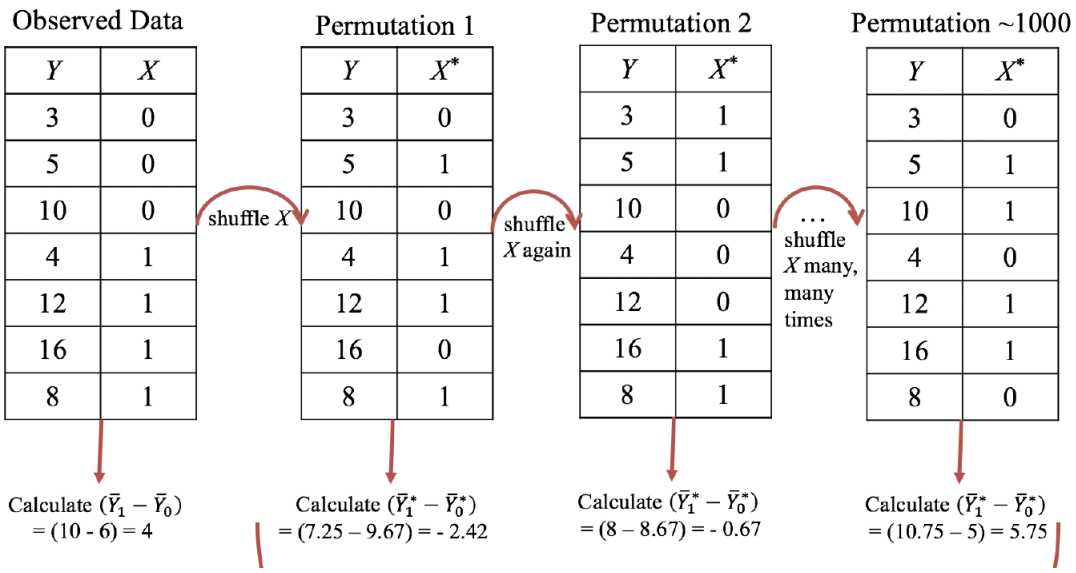


Figure 1: For each permutation, we calculate $\bar{Y}_0 - \bar{Y}_1$.

We then have a sampling distribution of the difference in means under permutation of the labels, $\bar{Y}_0 - \bar{Y}_1$. Then we can calculate empirical p-values, etc. This is independent of assuming any distribution for $\bar{Y}_0 - \bar{Y}_1$.

Differences between bootstrapping and permutation tests

- Goal: Bootstrapping estimates things, and permutation test is done to test a specific hypothesis. So, hypothesis testing cannot give you a confidence interval; it can only say whether

something is sensitive to group labeling.

- Implementation: Bootstrapping is done using an empirical sample from the real distribution, so effectively it is done assuming H_A is true. On the other hand, permutation testing is a hypothesis test and is performed strickly assuming H_0 is true.

Lecture 11: Diagnostics and Transformations

We would like to come up with diagnostics for (multi)linear regression. This means: how do we know all the assumptions for (multi)linear regression were satisfied?

Checking the assumptions for linear regression

1. Linearity: The means of subpopulations fall along a line. **Check**: Plot a scatterplot of the residuals ϵ versus fitted $\hat{y}$ and see if there are any correlations. **Fixes**: Consider transformations of either Y or X to make the fit more linear, or add interactions. Or, we could use polynomial regression, or other generalized models, such as linear regression on X^2-type terms.

2. Constant variance: The variances of subpopulations are equal no matter where we are along the line. **Check**: Plot a scatterplot of the residuals ϵ versus the fitted values $\hat{y}$. If the "size" of the scattering changes with y, then the variance is not constant! **Fixes**: Consider a transformation of Y to make the distribution skinnier for regions in which the variance is high, or use **weighted regression** where each observation is weighted inversely to its variance.

3. Normality: The subpopulations are distributed normally around the estimated mean. **Check**: Plot a QQ-plot or a histogram of the residuals; QQ-plot should be linear. **Fixes**: Transform Y or ignore the violation if not too bad.

4. Independence: the observations' responses are independent. **Check**: Am I missing any predictors? **Fixes**: If there are some correlations, we could add more predictors or use multilevel (hierarchical) models.

Usually, nonlinearities are the biggest issue, then independence, then constant variance, and lastly normality. There are also other issues, such as outliers and sever multicollinearity between the predictors.

Some tips

1. The response is most important. Transform it first, if needed.

2. Worry about the predictors later.

3. Use log-transforms to take care of big difference in order-of-magnitude. Logarithms make big things small and small things big.

Lecture 12: Categorical Predictors

We would like to know how to incorporate categorical predictors into a regression model. First, let's consider **binary predictors**, which can take only one value. If that is the only predictor X, then we write

$$Y = \hat{\beta}_0 + \hat{\beta}_1 X \tag{6}$$

which means that $\hat{\beta}_1$ estimates the difference in means between the $X = 1$ and $X = 0$ reference group. The resulting t-test for H_0: $\beta_1 = 0$ is mathematically equivalent ot the pooled 2-sample t-test.

Categorical predictors with more than 2 categories

What about categories which are more complicated than simple binary classifiers? If the classifier has K categories, we handle this by creating $K - 1$ binary (or dummy) variables, which can take the values 0 or 1, that recreate the groups. Then to determine if there are any differences among the K groups, we just perform the F-test. There are $K-1$ binary variables because the "default" option, when all $X_i = 0$, $1 \leq i \leq K_1$, is counted as a category itself. No two X_i can be 1 at the same time, at least in the dataset (but of course, you could see what happens mathematically based on the regression).

We can also combine regression on categorical *and* continuous predictors.

Are the categorical predictors important?

Maybe we want to know if adding categorical predictors was useful at all. To determine this, we can't just use the collection of associated t-tests for all the categorical predictors X_i (like hot dog, hamburger, sandwich) since actually they should all be grouped together.

To determine whether adding categorical variables is important, we can use an **extra sum of squares F-test**. Suppose the small model has p_1 predictors and the big model has p_2 predictors. We want to see if the extra $p_2 - p_1$ predictors were useful. The null hypothesis is

$$H_0 : \beta_{p_1+1} = \cdots \beta_{p_2} = 0. \tag{7}$$

The alternative is

$$H_A : \text{Some combination of the extra } \beta_j \text{ provides explanatory power.} \tag{8}$$

The **ESS F-test statistic** is given by

$$F = \frac{(\text{SSE}_1 - \text{SSE}_2)/(p_2 - p_1)}{\text{SSE}_2/(n - p_2 - 1)} \tag{9}$$

since $n - p_2 - 1$ is the df of the residuals in the larger model. In `R`, we just run an ANOVA test, which gives us the F statistic. See the notes.

Interactions

The **interaction effect** of two predictors is that the effect of X_1 on the response Y may depend on the value of another predictor X_2. There is an easy way to implement this in `R`; see the notes.

Lecture 13: Polynomials and Smoothers

If the data is nonlinear, one thing we can do is apply a nonlinear transformation on either the predictors or the response and see if the resulting fit is linear. However, we could also fit the original (non-transformed) data to a nonlinear fit, such as a polynomial regression. Or, we could implement a non-parametric, nonlinear fit such as a "running average" model.

There is a nice way to implement, for example, **quadratic models** in `R`. See the notes. To interpret things, you say "the relationship betwen Y and X is different depending on the value of X," or something like that. You can also fit higher-order polynomials but run the risk of overfitting. You can tell whether you overfit with certain measures of goodness-of-fit, like AIC, which account for the number of parameters. Or, you can regularize the irrelevant predictors away with techniques like LASSO.

Non-parametric smoothing

Another way to handle things is to use nonparametric methods. Let's meet an example of **non-parametric smoothing**: the famous **k-nearest neighbors**, or **kNN**, method. This is an example of a **moving average** method.

Basically, what happens is that to make a prediction $\hat{y}(\mathbf{X})$ you take the k nearest neighbors to $\mathbf{X}$, average their y values, and set that as the prediction $\hat{y}(\mathbf{X})$. This gives stepped responses and therefore people like to use other smoothers, like Gaussian coarse-graining, to smooth out the response.

A sophisticated method of implementing kNN is called **LOESS**. It weights the observations x_i in the neighborhood of a desired x^* using a **tricubic weighting** and then does some kind of fancy fit with the weighted least squares. The upshot is that it always gives smooth graphs.

Lecture 15: Model Comparison and Selection

We'd like to know how to compare models; specifically models with varying numbers of predictors. How do we penalize models which have more predictors?

Adjusted R^2

The definition of **adjusted** R^2 is

$$\text{Adjusted } R^2 = 1 - (1 - R^2)\left(\frac{n-1}{n-p-1}\right) \tag{10}$$

where p is the number of predictors and n is the sample size. As with the regular R^2, the adjusted R^2 can be negative.

Other criteria

The most commonly used criteria to penalize models with more predictors are AIC and BIC. These penalize the negative of log-likelihood. They are defined as

$$\text{AIC} = n\log(\frac{\text{SSE}}{n}) + 2p, \text{BIC} = n\log(\frac{\text{SSE}}{n}) + p\log n. \tag{11}$$

We pick the model with the lowest AIC or BIC.

Sequential variable selection

There are two directions of sequential variable selection: forwards and backwards.

Forwards sequential variable selection

1. Start with the simplest model (usually intercept-only), $E(Y|X) = \beta_0 = \text{const}$.
2. Consider all models with one more term added.
3. For each calculate some criterion, such as F-statistic, AIC, BIC, adj-R^2, etc.
4. Include the term with the best result for the criterion. For example, the lowest AIC.
5. Iterate steps 3 and 4 until no more variables can be added.

This can be implemented easily in `R`. See the notes.

Backwards sequential variable selection

1. Start with all predictors in the model (possibly interactions and polynomial terms as well).
2. Consider all models with one term removed.
3. For each calculate some criterion, such as F-statistic, AIC, BIC, adj-R^2.
4. Remove the term which has the least effect on the model, i.e. remove the term which, when removed, did not change how good the fit was by very much.
5. Iterate steps 3 and 4 until no more variables can be removed.

This is implemented easily in `R`. See the notes.

Stepwise variable selection

This method combines forward and backward variable selection.

1. Start with the simplest model (usually intercept-only), $E(Y|X) = \beta_0 = \text{const}$.
2. Consider all models with one more term added *or* one term eliminated. (At the first step, obviously the latter is impossible.)
3. For each calculate some criterion, such as F-statistic, AIC, BIC, adj-R^2, etc.
4. Include the term with the best result for the criterion. For example, the lowest AIC.
5. Iterate steps 3 and 4 until no more variables can be added or eliminated.

Lecture 16: Cross Validation

Cross-validation refers to testing the predictive model on something other than the training data. Often we do this by splitting the data into the **training set** and the **test set**.

If many models are being considered, we may also introduce a **validation set** which is used to compare model performances. For example, we may

1. Split the original data into train and test sets (80-20)
2. Split the training set into train and validation sets (64-16)

The best model is selection via validation, and then that best model is used on the test data.

Cross-validation

Actually, it can be dangerous to have a *single* test set, because we risk overfitting to the single training set. Let us study k-fold cross-validation.

1. Shuffle the dataset randomly.
2. Split the dataset into k groups.
3. For each unique group, assume it is the test set and the other $k-1$ groups are the training set. Fit the model on the training set and evaluate on the test set. Retain the evaluation score.
4. We end up with a list of k evaluation scores.

Then we can average the evaluation scores to get a better idea of how well the model performs.

Lecture 17: Ridge and Lasso

Ridge and Lasso are different types of **penalized regression**, which penalize us for having large values of the coefficients. Let the original penalty be

$$\text{SSE} = (\mathbf{y} - \mathbf{X} \cdot \beta)^T \cdot (\mathbf{y} - \mathbf{X} \cdot \beta). \tag{12}$$

$$\textbf{Ridge} : \arg \min_{\beta} \left(\text{SSE} + \lambda \sum_j \beta_j^2 \right) \tag{13}$$

The reason this is called "ridge" is due to the form of the solution for β, in which a "ridge" of λs are added to the diagonal of a matrix before inverting:

$$\beta_{\text{ridge}} = (\mathbf{X}^T\mathbf{X} + \lambda \mathbf{1})^{-1}\mathbf{X}^T\mathbf{y}. \tag{14}$$

In `R`, this is implemented using the `lm.ridge` command of the `MASS` package, or alternatively the `glmnet` package.

On the other hand, LASSO uses an absolute value penalty term, which looks like

$$\textbf{LASSO:} \arg\min_{\beta} \left(\text{SSE} + \lambda \sum_j |\beta_j| \right). \tag{15}$$

In fact, β_0 is invariant under LASSO or ridge!

Comparing LASSO and ridge

Both methods shrink the β_j towards zero. However, Lasso will often shrink the β_j to *exactly* zero, so it is often easier to interpret. This is a kind of variable selection. The following diagram illustrates why this is the case.

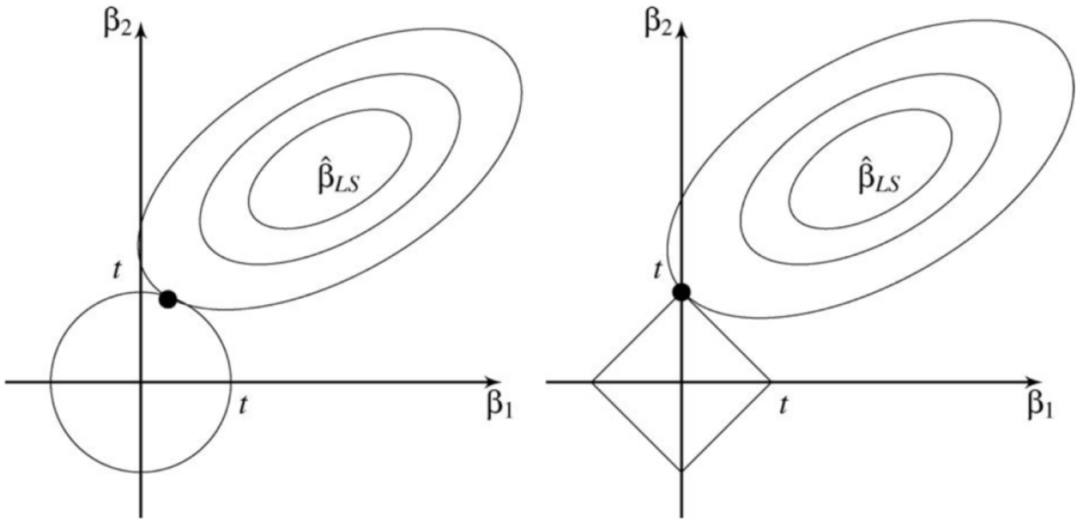


Figure 2: Ridge on left, lasso on right.

Lecture 18: Simulation

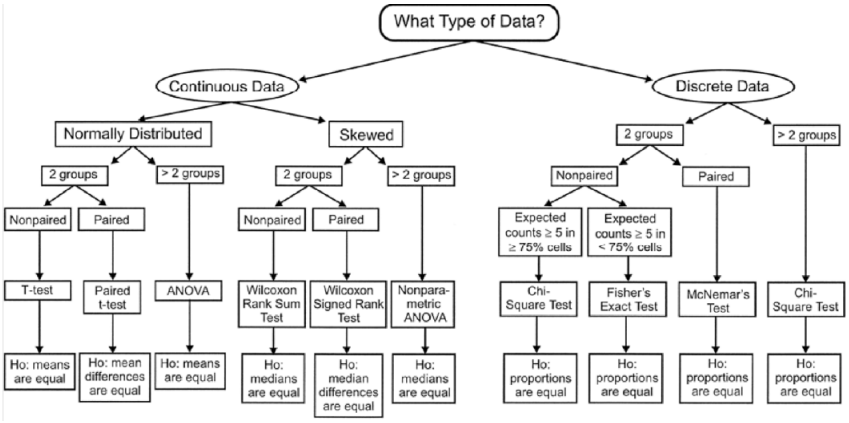
Sometimes we would like to run simulations to model what will actually happen. For example, we do this all the time in Stat 131: Times Series homework.

> Sequential Variable Selection: does really well when there are **only a few strong predictors**. But is *slow*.
>
> Ridge: does really well when there are **a lot of weak to moderately strong predictors**.
>
> Lasso: does really well when there is **a mix of useless predictors and moderate to strong predictors**.

Figure 3: Conclusions from this useless lecture.

Lecture 19: Decision Trees and Regression Trees

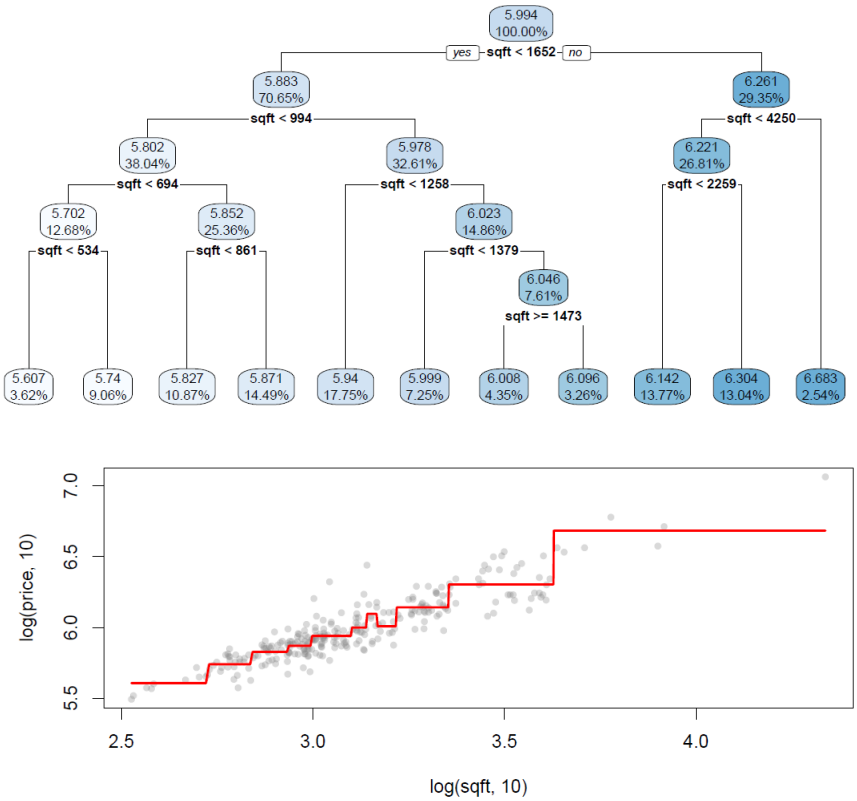
Here is an example of a **decision tree** – one that may actually be useful.



Regression trees

Decision trees can easily be constructed for categorical responses, but what about continuous responses? We combine the idea of decision trees and regression. The new model is called a **regression tree**.

An elementary way to implement a regression tree is to split the X-interval up into little pieces and approximate y as constant on each of those pieces. This is illustrated below.



The loss function for a *single split* is something like

$$\text{SSE} = \sum_{i \in R_1} (Y_i - \bar{Y}_1)^2 + \sum_{j \in R_2} (Y_j - \bar{Y}_2)^2. \tag{16}$$

We have to implement this kind of thing every time we split. $R_{1,2}$ are the two different regions and their boundary is called the **splitting threshold**.

Tuning trees

The **complexity** of a regression tree is measured by its **depth** and **size**, which is the number of terminal "leaves" in a tree. Once a large tree is created, we might want to "prune" it by removing splits. To do so, we use the **cost complexity** measure, which is

$$\text{Cost complexity} = \text{SSE} + \alpha |T|, \text{ where } |T| = \text{number of leaves.} \tag{17}$$

In `R`, α is called `cp`.

Lecture 20: Random Forests

The problem with trees on continuous data is that the predictions will have lots of steps. The derivative of the prediction surface will have δ-function spikes or something.

We would like to make use of the computational efficiency of trees, but smooth out the prediction surface. To do so, we use **random forests**.

Bagging

The approach to making a random forest is as follows: on the training data, we bootstrap resample over and over and refit the tree model to the bootstrapped data each time. Then, we average over all the bootstrapped trees. This is called **b**ootstrap **agg**regat**ing**, or **bagging**.

This general idea of combining many models to get a single final model is called an **ensemble** approach. There are two general approaches:

1. Boosting: Build a final ensemble model via underfit models, and have each model's residuals iteratively predicted by the next model to reduce bias.
2. Bagging: Fit lots of complex (and possibly overfit) models. These models have high variance; when we average them, we reduce the variance.

In the context of random forests, decision trees are called the **base model**.

What's random about a random forest?

Random forests *do* involve bagging, but each time we choose a bootstrapped sample, we also choose a subset of the predictors on which to do a decision tree. In `R`, we can choose how many predictors we want to split over at each node, etc.

Lecture 21: Weighted and Robust Regression

One of the assumptions in linear regression is **homoscedasticity**, or constant variance of the residuals. What about **heteroscedasticity**? Besides transformation methods, we can implement more exotic methods to attack this problem.

The **general heteroscedastic model** is like a linear regression, except the residuals are independent but of varying size,

$$\mathbf{Y} = \mathbf{X} \cdot \boldsymbol{\beta} + \boldsymbol{\epsilon}, \text{ where } \boldsymbol{\epsilon} \sim \text{MVN}(0, \boldsymbol{\Sigma}) \tag{18}$$

and $\boldsymbol{\Sigma} = \text{diag}(\sigma_1^2, \cdots, \sigma_n^2)$. These are called **robust standard errors** or **sandwich estimators of variance**.

Weighted least squares regression

Suppose, as before, that the variances of the residuals are unequal. However, we don't really want to introduce a distribution σ_n^2 of variances of the residuals. Let's choose a different method: weight each point differently.

Denote the weighting of point i with the function w_i. The cost function is

$$\mathcal{L} = \sum_i^n w_i \Big(Y_i - (\beta_0 + \boldsymbol{\beta} \cdot \mathbf{X}_i) \Big)^2 \tag{19}$$

This raises the question of how to determine the functional form w_i. What is its dependence on i?

If not already suggested by design, we can fit the ordinary least-squares regression and examine the sizes of the residuals against i. If the magnitudes increase proportionally to one of the X_i, perhaps we can fit a weighted least-squares with weight $w_i \sim X_i^{-1}$ or $w_i \sim X_i^{-2}$.

Another method is to use

$$w_i \sim \frac{1}{(Y_i - \hat{Y}_i)^2}, \tag{20}$$

fit the model, obtain the coefficients and weights, use the weights to recalculate the coefficients, and repeat the process iteratively until the model converges. This is implemented in the `MASS` package by function `rlm`.

Robust regression

A problem in regression is the presence of outliers. A method is called **robust** if it is insensitive to the presence of outliers. **Huber's method** depresses the effect of outliers by treating them with absolute error weight, rather than quadratic weight, in the cost function:

$$\begin{aligned} g(e_i) &= \frac{1}{2}e_i^2 \text{ if } |e_i| \le c \\ &= c|e_i| - \frac{1}{2}c^2 \text{if } |e_i| > c. \end{aligned}$$

Another kind of robust regression is **median regression**, which uses absolute error loss $g(e_i) = |e_i|$.

Lecture 22: Non-independent Data

We would like to extend the idea of regression to data which are not independent. We will do this in two ways: generalized least squares and hierarchical (mixed-effects) models.

Generalized least squares

The **GLS** assumption is that the residuals can be correlated. Namely,

$$\mathbf{Y} = \mathbf{X} \cdot \boldsymbol{\beta} + \boldsymbol{\epsilon} \tag{21}$$

where the residuals are correlated,

$$\epsilon \sim \text{MVN}(0, \sigma^2 \times \boldsymbol{\Sigma}). \tag{22}$$

Here, $\sigma^2 \in \mathbb{R}$ is a scalar and $\boldsymbol{\Sigma}$ is a matrix. This is the structure, for example, in AR(p) models. If we can decompose

$$\boldsymbol{\Sigma} = \mathbf{S}\mathbf{S}^T, \tag{23}$$

then we can solve the problem by using regular least squares on the transformed

$$\mathbf{Y}^* = \mathbf{S}^T\mathbf{Y} \text{ and } \mathbf{X}^* = \mathbf{S}^T\mathbf{X}. \tag{24}$$

To perform GLS in `R`, we use the `gls` command in the `nlme` package.

Hierarchical models

Hierarchical models are when there are some categorical predictors which clearly split the data into different "categories," such as a "county" predictor for the statistics of a state's population. If we are interested in the effects of the continuous predictors rather than the effects of the categorical predictors, we can let the slopes and intercepts w.r.t. the continuous predictors *depend* on the categorical predictors. This is a **hierarchical model**.

If we let the intercepts vary with categorical variable, we call this a **random-intercept model**; if we let the slopes and intercepts vary, we call this a **random-intercept-and-slope model**.

Lecture 23: More Mixed Models

Suppose we use, say, a random-intercept model, in which the intercepts α_j are pulled from the distribution

$$\alpha_j|\mu_\alpha, \sigma_\alpha^2 \sim \mathcal{N}(\mu_\alpha, \sigma_\alpha^2). \tag{25}$$

We can estimate the parameters $\mu_\alpha, \sigma_\alpha^2$ by using standard maximum likelihood estimators. Actually, it's a bit complicated so people use sophisticated methods to do so.

Lecture 24: Mixed Effects for Time

This is the dumbest lecture I have *ever* seen...

Lecture 25: Logistic Regression

The idea of **logistic regression** is that if the response variable is categorical, and can take values 0 or 1, we need the output of the regression to be between 0 and 1 (like a probability). How can we do this?

We introduce the **logistic function**

$$\text{logit}(x) = \log\left(\frac{x}{1-x}\right). \tag{26}$$

The logit function maps the interval $(0, 1)$ to the interval $(-\infty, \infty)$. Therefore, the **logistic regression** model is

$$\log\left(\frac{P(Y=1)}{1-P(Y=1)}\right) = \beta_0 + \boldsymbol{\beta}\cdot\mathbf{X} \tag{27}$$

or equivalently

$$P(Y = 1) = \left(1 + e^{-(\beta_0 + \boldsymbol{\beta} \cdot \mathbf{X})}\right)^{-1}. \tag{28}$$

The predictor variables can be quantitative or binary.

Odds ratio

The preferred way of interpreting the effect of a predictor on the response in a logistic regression model is called the **odds ratio**, which can be calculated for each predictor variable.

An odds ratio is the ratio between odds, and odds themselves are ratios. Yikes...

The below picture is from Wikipedia:

$$\exp(\beta_x) = \frac{P(Y = 1 \mid X = 1, Z_1, \ldots, Z_p)/P(Y = 0 \mid X = 1, Z_1, \ldots, Z_p)}{P(Y = 1 \mid X = 0, Z_1, \ldots, Z_p)/P(Y = 0 \mid X = 0, Z_1, \ldots, Z_p)}$$

Conclusion: the odds ratio is just $e^{\hat{\beta}_i}$, where β_i is the slope associated with predictor X_i.